Electrical Flows, Laplacian Matrices, and New Approaches to the Maximum Flow Problem

Aleksander Mądry
Maximum flow problem

**Input:** Directed graph $G$, integer capacities $u_e$, source $s$ and sink $t$

**Task:** Find a feasible $s$-$t$ flow of max value

**Value** = net flow out of $s$

- No overflow on arcs: $0 \leq f(e) \leq u(e)$
- No leaks at all $v \neq s, t$

Max flow value $F^* = 10$
Breaking the $O(n^{3/2})$ barrier

Undirected graphs and approx. answers ($O(n^{3/2})$ barrier still holds here)

[M ‘10]: Crude approx. of max flow value in close to linear time

[CKMST ‘11]: (1-\(\varepsilon\))-approx. to max flow in $\tilde{O}(n^{4/3}\varepsilon^{-3})$ time

[LSR ‘13, S ‘13, KLOS ‘14]: (1-\(\varepsilon\))-approx. in close to linear time

But: What about the directed and exact setting?

[M ‘13]: Exact $\tilde{O}(n^{10/7})=\tilde{O}(n^{1.43})$-time alg.

(n = # of vertices, $\tilde{O}()$ hides polylog factors)
Breaking the $O(n^{3/2})$ barrier

Undirected graphs and approx. answers ($O(n^{3/2})$ barrier still holds here)

[M ‘10]: Crude approx. of max flow value in close to linear time

[CKMST ‘11]: (1-ε)-approx. to max flow in $\tilde{O}(n^{4/3}ε^{-3})$ time

[LSR ‘13, S ‘13, KLOS ‘14]: (1-ε)-approx. in close to linear time

But: What about the directed and exact setting?

[M ‘13]: Exact $\tilde{O}(n^{10/7})=\tilde{O}(n^{1.43})$-time alg.

Today

($n =$ # of vertices, $\tilde{O}()$ hides polylog factors)
From electrical flows to exact directed max flow

From now on: All capacities are 1, $m=O(n)$ and the value $F^*$ of max flow is known
Why the progress on approx. undirected max flow does not apply to the exact directed case?

Tempting answer: Directed graphs are just different (for one, electrical flow is an undirected notion)

But: exact directed max flow reduces to exact undirected case

So, it is all about getting

Key obstacle: Gradient descent methods (like MWU) are inherently unable to deliver good enough accuracy
(Path-following) Interior-point method (IPM)
[Dikin ‘67, Karmarkar ‘84, Renegar ‘88,...]
A powerful framework for solving general LPs (and more)

LP: \[ \min c^T x \]
\[ \text{s.t. } Ax = b \]
\[ x \geq 0 \]

Idea: Take care of “hard” constraints by adding a “barrier” to the objective

“easy” constraints (use projection)

“hard” constraints
**Path-following) Interior-point method (IPM)**

[Dikin ‘67, Karmarkar ‘84, Renegar ‘88,...]

A powerful framework for solving general LPs (and more)

\[
\text{LP}(\mu): \min c^T x - \mu \sum_i \log x_i \\
\text{s.t. } Ax = b \\
x \geq 0
\]

**Idea:** Take care of “hard” constraints by adding a “barrier” to the objective

**Observe:** The barrier term enforces \( x \geq 0 \) implicitly

**Furthermore:** for large \( \mu \), \( \text{LP}(\mu) \) is easy to solve and

\[
\text{LP}(\mu) \rightarrow \text{original LP, as } \mu \rightarrow 0^+
\]

**Path-following routine:**

→ Start with (near-)optimal solution to \( \text{LP}(\mu) \) for large \( \mu > 0 \)
→ Gradually reduce \( \mu \) while maintaining the (near-)optimal solution to current \( \text{LP}(\mu) \)
(Path-following) Interior-point method (IPM)

[Dikin ‘67, Karmarkar ’84, Renegar ’88,...]

A powerful framework for solving general LPs (and more)

LP(μ): \( \min c^T x - \mu \sum_i \log x_i \)

s.t. \( Ax = b \)

\( x \geq 0 \)

Idea: Take care of “hard” constraints by adding a “barrier” to the objective

Observe: The barrier term enforces \( x \geq 0 \) implicitly

Path-following routine:

\( \rightarrow \) Maintain (near-)optimal solution

\( \rightarrow \) Repeat:

Set \( \mu' = (1-\delta)\mu \) and use Newton’s method to compute from \( x \)

(near-)optimal solution to \( \text{LP}(\mu') \)

Based on second-order approx.

\[ f(x+y) \approx f(x) + y^T \nabla f(x) + y^T H_f(x)y \]

+ projection on \( \ker(A) \)

Key point: Choosing step size \( \delta \) sufficiently small ensures \( x \) is close to optimum for \( \text{LP}(\mu') \) \( \rightarrow \) Newton’s method convergence very rapid
Path-following routine:

→ Start with (near-)optimal solution to LP(μ) for large μ>0

→ Gradually reduce μ (via Newton’s method) while maintaining the (near-)optimal solution to current LP(μ)

$\mathbf{P}_A = \{ \mathbf{x} \mid \mathbf{A}\mathbf{x} = \mathbf{b} \}$

$\mathbf{x}_c$ – analytic center

*central path* = optimal solutions to LP(μ) for all μ>0
Can we use IPM to get a faster max flow alg.?

**Conventional wisdom:** This will be too slow!

→ Each *Newton's step* = solving a linear system $O(n^\omega) = O(n^{2.373})$ time (prohibitive!)

**But:** When solving *flow problems* – only $\tilde{O}(m)$ time [DS ‘08]

**Fundamental question:** What is the number of iterations?

[Renegar ‘88]: $O(m^{1/2} \log \varepsilon^{-1})$

**Unfortunately:** This gives only an $\tilde{O}(m^{3/2})$-time algorithm

**Improve the $O(m^{1/2})$ bound?**
Although believed to be very suboptimal, its improvement is a major challenge
The Max Flow algorithm

(Self-contained, but can be seen as a variation on IPM)
From Max Flow to Min-cost Flow

Reduce **max flow** to **uncapacitated min-cost σ-flow problem**
From Max Flow to Min-cost Flow

Reduce **max flow** to uncapacitated min-cost $\sigma$-flow problem
From Max Flow to Min-cost Flow

Reduce max flow to uncapacitated min-cost $\sigma$-flow problem

Result: Feasibility $\rightarrow$ Optimization $+$ special structure
Solving Min-Cost Max Flow Instance

- Primal solution: $\sigma$-flow $f$
  (feasibility: all $f_e$ are $\geq 0$)

- Dual solution: embedding $y$ into real line
  (feasibility: all slacks $s_e$ are $\geq 0$)

“No arc is too stretched”

Our approach is **primal-dual**
Solving Min-Cost Max Flow Instance

Our approach is **primal-dual**

→ **Primal** solution: $\sigma$-flow $f$  
   (feasibility: all $f_e$ are $\geq 0$)

→ **Dual** solution: embedding $y$ into real line  
   (feasibility: all slacks $s_e$ are $\geq 0$)

"No arc is too stretched"
Solving Min-Cost Max Flow Instance

Our Goal:
Get \((f,y)\) with small duality gap \(\Sigma e f_e s_e\)

Our Approach: Iteratively improve maintained solution while enforcing an additional constraint

Centrality:
\[ f_e s_e \approx \mu, \text{ for all } e \]
(with \(\mu\) being progressively smaller)

“Make all arcs have similar contribution to the duality gap”
(Maintaining centrality = following the central path)
Taking an Improvement Step

So far, our approach is fairly standard

**Crucial Question:**
How to improve the quality of maintained solution?

**Key Ingredient:**
Use electrical flows
Taking an Improvement Step

Let \((f,y)\) be a (centered) primal-dual solution

**Key step:** Compute electrical \(\sigma\)-flow \(f^+\) with \(r_e := s_e / f_e\)

**Primal improvement:** Set \(f' := (1-\delta)f + \delta f^+\)

**Dual improvement:** Use voltages \(\varphi\) inducing \(f^+\) (via Ohm’s Law)
Set \(y' := y + \delta(1-\delta)^{-1} \varphi\)

**Can show:** When terms quadratic in \(\delta\) are ignored

\[ f_e' s_e' \approx (1-\delta) \mu = \mu' \]

for each \(e\)

(i.e., **duality gap** decreases by \((1-\delta)\) and **centrality** is preserved)

How big \(\delta\) can we take to have this approx. hold?
Lowerbounding $\delta$

Can show: $\delta^{-1}$ is bounded by $O(|\rho|_4)$ where $\rho_e := |f_e^+|/f_e$

$|\rho|_4$ measures how different $f^+$ and $f$ are

How to bound $|\rho|_4$?

Idea: Bound $|\rho|_2 \geq |\rho|_4$ instead
**Lowerbounding δ**

*Can show:*  
\( \delta^{-1} \) is bounded by \( O(|\rho|_4) \)  
where \( \rho_e := |f_e^+|/f_e \)

\( |\rho|_4 \) measures how different \( f^+ \) and \( f \) are  

How to bound \( |\rho|_2 \)?  
\( (|\rho|_2 \geq |\rho|_4) \)

**Centrality:**  
Tying \( |\rho|_2 \) to \( E(f^+) \)  
\[ f_e s_e \approx \mu \rightarrow r_e = s_e/f_e \approx \mu/(f_e)^2 \]  
\[ \downarrow \]  
\[ E(f^+) \approx \mu (|\rho|_2)^2 \]
Lowerbounding $\delta$

Can show: $\delta^{-1}$ is bounded by $O(|\rho|_4)$ where $\rho_e := |f_e^+|/f_e$

How to bound $|\rho|_2$? ($|\rho|_2 \geq |\rho|_4$)

Centrality: Tying $|\rho|_2$ to $E(f^+)$

$f_e s_e \approx \mu \rightarrow r_e = s_e/f_e \approx \mu/(f_e)^2$

$E(f^+) = \Sigma_e r_e (f_e^+)^2 \approx \Sigma_e \mu (f_e^+/f_e)^2 = \mu \Sigma_e (\rho_e)^2 = \mu (|\rho|_2)^2$

So, we can focus on bounding $E(f^+)$
Lowerbounding $\delta$

Can show: $\delta^{-1}$ is bounded by $O(|\rho|_4)$ where $\rho_e := |f^+_e|/f_e$.

How to bound $|\rho|_2$? ($|\rho|_2 \geq |\rho|_4$)

How to bound $E(f^+)$? ($E(f^+) \approx \mu (|\rho|_2)^2$)

Idea: Use energy-bounding argument we used in the undirected case.

Claim: $E(f^+) \leq \mu m$

Proof: Note that $E(f) = \sum_e r_e (f_e)^2 \approx \sum_e \mu (f_e/f_e)^2$.
**Lowerbounding $\delta$**

**Can show:**

$\delta^{-1}$ is bounded by $O(|\rho|_4)$

where $\rho_e := |f_e^+|/f_e$

$|\rho|_4$ measures how different $f^+$ and $f$ are

How to bound $|\rho|_2$?

$|\rho|_2 \geq |\rho|_4$

How to bound $E(f^+)$?

$(E(f^+)) \approx \mu (|\rho|_2^2)$

**Idea:** Use energy-bounding argument we used in the undirected case

**Claim:** $E(f^+) \leq \mu m$

**Proof:** Note that $E(f) = \sum_o r_o (f_o)^2 \approx \mu \sum_o (f_o/f_e)^2 = \mu \sum_o 1 = \mu m$

**Result:** Bounding $\delta^{-1} \leq |\rho|_4 \leq |\rho|_2 \leq (E(f^+)/\mu)^{1/2} \leq m^{1/2}$

$E(f^+) \leq E(f) \approx \mu m$

This recovers the canonical $O(m^{1/2})$-iterations bound for general IPMs and gives the $\tilde{O}(m^{3/2} \log U)$ algorithm
Going beyond $\Omega(m^{1/2})$ barrier

Our reasoning before: $\delta^{-1} \leq |\rho|_4 \leq |\rho|_2 \leq m^{1/2}$

Essentially tight in our framework
Going beyond $\Omega(m^{1/2})$ barrier

Our reasoning before: $\delta^{-1} \leq |\rho|_4 \leq |\rho|_2 \leq m^{1/2}$

When does $|\rho|_4 \approx |\rho|_2$?

This part we need to improve
Going beyond $\Omega(m^{1/2})$ barrier

Our reasoning before: $\delta^{-1} \leq |\rho|_4 \leq |\rho|_2 \leq m^{1/2}$

When does $|\rho|_4 \approx |\rho|_2$?

**Answer:** If most of the norm of $\rho$ is focused on only a few coordinates.

Translated to our setting: $|\rho|_4 \approx |\rho|_2$ if most of the energy of $f^+$ is contributed by only a few arcs.

Can this happen? Unfortunately, yes.

Contributes most of the energy

$\approx n^{1/2}$
Going beyond $\Omega(m^{1/2})$ barrier

Our reasoning before: $\delta^{-1} \leq |\rho|_4 \leq |\rho|_2 \leq m^{1/2}$

When does $|\rho|_4 \approx |\rho|_2$?

Answer: If most of the norm of $\rho$ is focused on only a few coordinates.

Translated to our setting: $|\rho|_4 \approx |\rho|_2$ if most of the energy of $f^+$ is contributed by only a few arcs.

Can this happen? Unfortunately, yes (in principle, tight)

This is the only part where unit-capacity assumption is needed.

Method: Very careful perturbation of the solution + certain preconditioning.
Going beyond $\Omega(m^{1/2})$ barrier

**Problematic case:** When most of the energy of $f^+$ is contributed by only a few arcs

How can we ensure that this is not the case?

We already faced such problems in the undirected setting!
Going beyond $\Omega(m^{1/2})$ barrier

**Problematic case:** When most of the energy of $f^+$ is contributed by only a few arcs

How can we ensure that this is not the case?

We already faced such problems in the undirected setting!

**Our approach then:** Keep removing high-energy edges

**To show this works:** Used the energy of the electrical flow as a potential function

- Energy can only increase and obeys global upper bound
- Each time removal happens $\rightarrow$ energy increases by a lot

**Problems:** In our framework, arc removal is too drastic and the energy of $f^+$ is highly non-monotone
Going beyond $\Omega(m^{1/2})$ barrier

How to deal with these problems?

→ Enforce a **stronger** condition than just that $|\rho|_4$ is small ("smoothness": restrict energy contributions of arc subsets)

**Key fact:** $f^+$ smooth $\rightarrow$ energy does not change too much (so, energy becomes a good potential function again)

→ To enforce this, keep **stretching** the offending arcs (stretch = increase length by $s_e$ - this doubles the resistance $r_e = s_e/f_e$)

As long as $s_e$ is small for stretched arcs, the resulting perturbation of lengths can be corrected at the end

**Remaining question:** How to handle arcs with large $s_e$?
Going beyond $\Omega(m^{1/2})$ barrier

Observation: As $f_es_e \approx \mu$, large $s_e \rightarrow$ small flow $f_e$ and thus $r_e = s_e/f_e \approx \mu/f_e^2$ is pretty large

→ For such arcs: contributing a lot of energy implies high effective resistance

Idea: Precondition $(f,y)$ so as no arc has too high effect. resist.
Going beyond $\Omega(m^{1/2})$ barrier

**Observation:** As $f_e s_e \approx \mu$, large $s_e \rightarrow$ small flow $f_e$
and thus $r_e = s_e / f_e \approx \mu / f_e^2$ is pretty large

→ **For such arcs:** contributing a lot of energy implies high **effective** resistance

**Idea:** Precondition $(f, y)$ so as no arc has too high effect. resist.

![Auxiliary star graph](image)
Going beyond $\Omega(m^{1/2})$ barrier

Observation: As $f_e s_e \approx \mu$, large $s_e \rightarrow$ small flow $f_e$ and thus $r_e = s_e / f_e \approx \mu / f_e^2$ is pretty large

→ For such arcs: contributing a lot of energy implies high effective resistance

Idea: Precondition $(f,y)$ so as no arc has too high effect. resist.

Can show: After doing that, no arc with large $s_e$ contributes significant portion of energy

Furthermore: The flow routed over auxiliary arcs is small enough that it can be rerouted without destroying our overall progress

Putting these two techniques together + some work: $\tilde{O}(m^{3/7})$-iterations convergence follows
Conclusions and the Bigger Picture
Maximum Flows and Electrical Flows

Elect. flows + IPMs $\rightarrow$ A powerful new approach to max flow

Can this lead to a nearly-linear time algorithm for the exact directed max flow?

We seem to have the “critical mass” of ideas

Elect. flows = next generation of “spectral” tools?

- Better “spectral” graph partitioning,
- Algorithmic grasp of random walks,
- ...
**Grand challenge:** Can we make algorithmic graph theory run in nearly-linear time?

**New “recipe”:** Fast alg. for **combinatorial** problems via **linear-algebraic** tools + **continuous opt.** methods

How about applying this framework to other graph problems that “got stuck” at $O(n^{3/2})$? (min-cost flow, general matchings, negative-lengths shortest path...)

**Second-order/IPM-like methods:** the next frontier for fast (graph) algorithms?
Max Flow and Interior-Point Methods

Contributing back: Max flow and electrical flows as a lens for analyzing general IPMs?

Our techniques can be lifted to the general LP setting

We can solve any LP within $\tilde{O}(m^{3/7}L)$ iterations

But: this involves perturbing of this LP

Some (seemingly) new elements of our approach:
• Better grasp of $\ell_2$ vs. $\ell_4$ interplay wrt the step size $\delta$
• Perturbing the central path when needed
• Usage of non-local convergence arguments

Can this lead to breaking the $\Omega(m^{1/2})$ barrier for all LPs?

[Lee Sidford ‘14]: $\tilde{O}(\text{rank}(A)^{1/2})$ iteration bound
Unifying the landscape of algorithmic graph theory

Research on graph algorithms is quite fragmented.

Can we establish a more unified picture of the field?

Complexity theory for sub-quadratic time algorithms?

Which graph problems can be efficiently reduced to other problems?
Study of “max flow”-hardness/completeness?
Bridging the Combinatorial and the Continuous

paths, trees, partitions, routings, matchings, data structures...  \[ \rightarrow \]  matrices, eigenvalues, linear systems, gradients, convex sets...

**Powerful approach:** Exploiting the interplay of the two worlds

Some other early “success stories” of this approach:
- Spectral graph theory aka the “eigenvalue connection”
- Fast SDD/Laplacian system solvers
- Graph sparsification, random spanning tree generation

...and this is just the beginning!
Thank you

Questions?